```
/*
  Relay- and Optocoupler Adapter for the PC-Printer-Port

  needs no external Power-Supply


  begin of programming: 08.01.1993
  last change:          17.02.1993
  qualification:        17.02.1993

  Dr.T.Aschmoneit                                                    */


/* function-prototypes */

void main ();
void RelayOn(unsigned char num);
void RelayOff(unsigned char num);
unsigned char OptoIn(void);

#include <stdio.h>
#include <dos.h>

unsigned x;
unsigned i;

unsigned far *printer = 0x00400008;


void main()

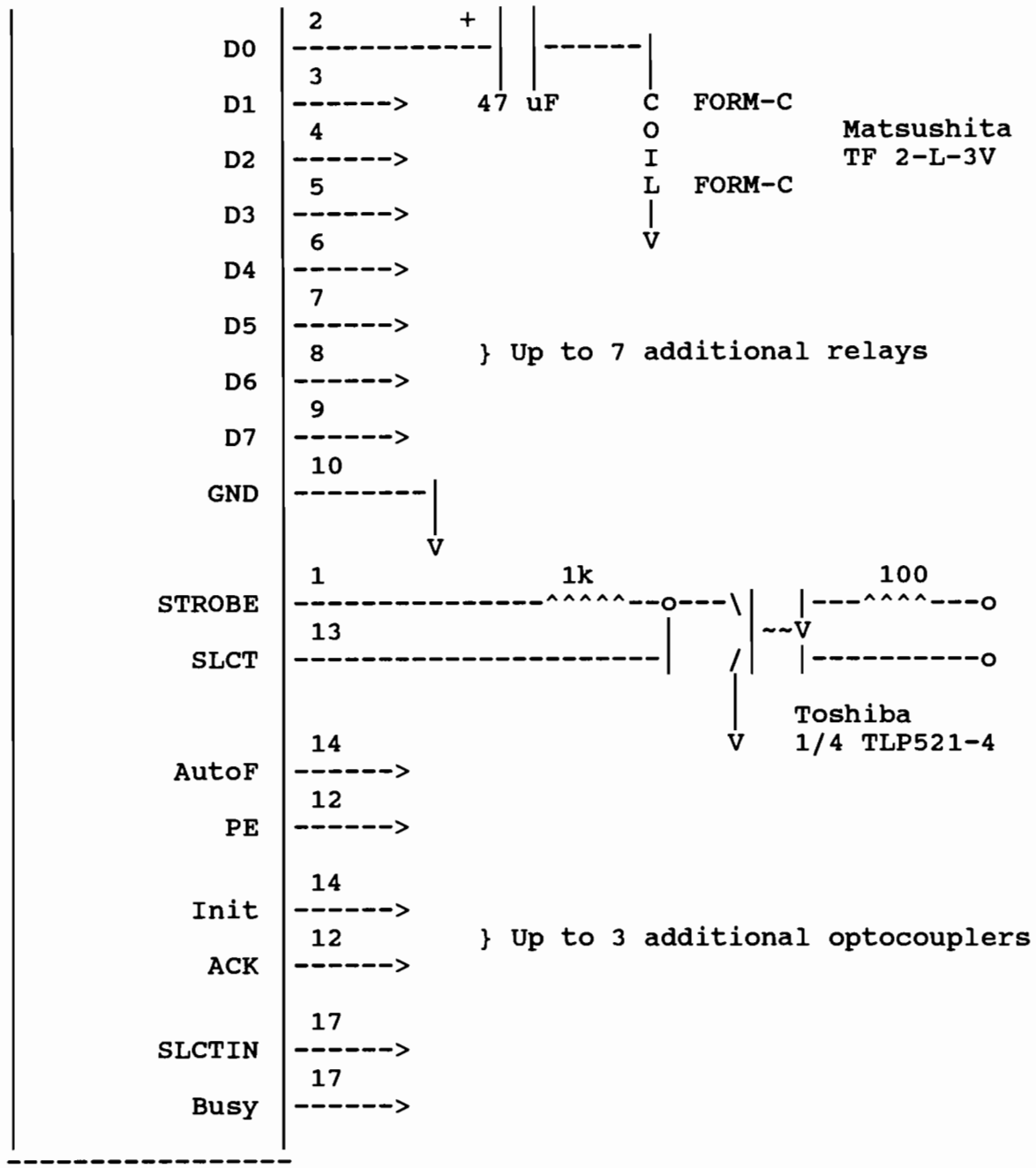

  RelayOn(0);

  while (1==1) {
    printf("%x \n",OptoIn());
  }
}


void RelayOn(unsigned char num)  /* num Ÿ [0,7] */
{ outportb(*printer,(1<<num) | inportb(*printer)); }

void RelayOff(unsigned char num)  /* num Ÿ [0,7] */
{ outportb(*printer,~(1<<num) & inportb(*printer)); }

unsigned char OptoIn(void)
{ outportb(*printer+2,4);     /* initialization */
  return ((inportb(*printer+1) >> 4) ^ 7); }
```


PC Centronics
Printer Port
D0
D1
D2
D3
D4
D5
D6
D7
GND
STROBE
SLCT
AutoF
PE
Init
ACK
SLCTIN
Busy
47 uF
COIL
FORM-C
FORM-C
Matsushita
TF 2-L-3V
} Up to 7 additional relays
1k
100
Toshiba
1/4 TLP521-4
} Up to 3 additional optocouplers

Relay- and optocoupler interface for the PC printer port

The circuit in Fig.1 shows the connection of eight latching relays and four optocouplers to the IBM-PC/AT Centronics port.

It's a very cheap and simple way for isolated (1500V) controlling and sensing signals with a PC, without the need of an additional power supply. The single coil latching relays are directly driven by the TTL outputs. The collector-current for the photo-transistor is also delivered by a TTL output.

Only eight lines of C code are necessary to control this circuit. First a far-pointer to the printer port's adress in the BIOS RAM must be declared. The functions "RelayOn" and "RelayOff" put the relay, which is defined in the argument on or off. The function "OptoIn" returns a nibble. Its bits are set or reset corresponding to the states of the four optocoupler-inputs.

Listing 1 Three functions for handling the adapter

```
unsigned far *printer = 0x00400008;

void RelayOn(unsigned char num)  /* num = 0..7 */
{ outportb(*printer,(1<<num) | inportb(*printer)); }

void RelayOff(unsigned char num)  /* num = 0..7 */
{ outportb(*printer,~(1<<num) & inportb(*printer)); }

unsigned char OptoIn(void) /* returns the input nibble*/
{ outportb(*printer+2,4);  /* initialization */
  return ((inportb(*printer+1) >> 4) ^ 7); }
```